

1. Bezpieczeństwo w monitoringu

Piotr Brudło

Politechnika Gdańska,

Wydział Elektroniki, Telekomunikacji i Informatyki,

Katedra Architektury Systemów Komputerowych

e-mail: pebrd@eti.pg.gda.pl

Streszczenie

Zaproponowano szereg algorytmów realizujących aspekty bezpieczeństwa zorientowane na aplikacje monitorujące. W znaczącej części pozwolą one na zastąpienie pracy czynnika ludzkiego przy przeglądaniu lub analizie zapisów video z monitorowanych miejsc lub obszarów. Ze względu na szeroki potencjalny obszar, zakres zastosowań praktycznych oraz potencjalne efekty natury ekonomicznej u potencjalnych klientów spodziewane jest szerokie wykorzystanie tego typu rozwiązań w automatycznym monitoringu zarówno typu offline jak i online.

Słowa kluczowe: aktywny monitoring, bezpieczeństwo, przekształcenia obrazu, formaty video

1.1. Problemy realizacyjne w aktywnym monitoringu video

W systemowych rozwiązaniach informatycznych w zakresie aktywnego monitoringu podstawowym problemem realizacyjnym jest analiza obrazów video w celu wykrycia sytuacji potencjalnie niebezpiecznych lub nietypowych. W automatycznych systemach nadzoru należy sklasyfikować sytuacje, efekty, obiekty, zdarzenia, etc. będące przedmiotem potencjalnego zainteresowania, a następnie zastosować odpowiednie algorytmy poszukujące ich w strumieniach (*online*) lub zapisach (*offline*) video.

1.1.1. Wykrywanie niebezpiecznych przedmiotów lub obiektów w obrazie

W przypadku obserwacji lub też rejestracji transmisji monitoringu z kamer to człowiek podejmuje decyzje dotyczące ewentualnej zakwalifikowania obserwowanego obrazu do sytuacji interesujących. Stąd też w tego typu zastosowaniach konieczna jest aktywna obserwacja obrazów z kamer przez służby nadzoru. W wypadku zastosowania systemu Comcute istnieje możliwość automatycznego wykrywania potencjalnie niebezpiecznych (niepożądanych) przedmiotów bądź obiektów pojawiających się w obserwowanym obszarze.

Rejestracja wideo może być podzielona na klatki, każda klatka może zostać wysłana do przetwarzania u określonego internauty. Proponuje się opracowanie bazy wzorców niepożądaných obiektów lub przedmiotów oraz ich wykrywanie. Dla przykładu przedmiotami potencjalnie niebezpiecznymi mogą być: duże torby lub walizki (z potencjalnie niebezpieczną zawartością), noże, czy broń palna. Szersze zastosowania mogą dotyczyć np. wykrywania ciężarówek (tirów) na drogach lub miejscach, gdzie nie powinny się one poruszać lub pojawiać, albo też samochodów na drogach z zakazem ruchu. Zadaniem aplikacji jest wykrycie na obrazie interesującego obiektu z bazy wzorców oraz wysłanie odpowiedniego sygnału dla służb, aby zdecydowały o ostatecznej klasyfikacji danego analizowanego przypadku.

1.1.2. Wykrywanie anomalii natury ogólnej w obrazie z kamer

W praktyce monitorowania zwykle poszczególne klatki z zapisu monitoringu statystycznie niewiele się od siebie różnią. Większość planu obrazu zajmuje tło, natomiast obiekty, które pojawiają się w zakresie wizyjnym, zajmują stosunkowo niewielki procent powierzchni. Celem aplikacji jest wykrywanie sytuacji, gdy zmiany obserwowanego tła są ponadprzeciętne. Dla statycznego obrazu można po wstępnym odfiltrowaniu wyznaczyć określony wzorzec tła (policzyć np. sygnaturę lub model widmowy). Wzorzec ten po korelacji z obrazem aktualnym (przetworzonym) pokazuje skalę potencjalnych zmian. Jeżeli zmiany te są zbyt duże, to zgłaszany jest sygnał do obsługi. Jako przykład zastosowania można podać wykrycie sytuacji zasnucia obrazu parkingu, lotniska czy drogi np.: gęstą mgłą, ale również pojawienie się np. dymu zasłaniającego znaczną część widoku kamery, czy zasłonięcie obiektywu kamery np. ręką lub innym przedmiotem. W przypadku trywialnym będzie to szybkie wykrycie odłączenia kamery (transmisja czarnego tła). W przypadku monitorowania przestrzeni, gdzie praktycznie nic nie powinno się pojawiać (ochrona obiektów), na podstawie różnic w obrazach można otrzymać sygnał o pojawieniu się nietypowych zmian (bez klasyfikacji co to jest).

1.1.3. Wykrywanie dynamicznych zmian w obrazach monitorujących

W przypadku dużych i dynamicznych zmian pomiędzy klatkami, zmiany te świadczyć mogą o potencjalnie interesujących sytuacjach. Przy wykorzystaniu operacji korelacji (porównania ze sobą) sekwencji kolejnych klatek i zaobserwowaniu różnic można wykryć potencjalnie interesujące sytuacje, takie jak: szybkie rozprzestrzenianie się dymu we fragmencie obrazu, wybuch obiektu w tle, rozszerzający się pożar, zbyt szybko poruszający się obiekt (np. samochód czy motocykl). W aplikacji wymagane jest wykonanie szeregu testów i prób, aby dla określonej kamery (tło, środowisko, dynamizm zmian) określić wartości progowe dla zgłaszania sygnału o wykryciu potencjalnych odstępstw od normy. Aplikacja może działać zarówno *online* (sygnały pojawiają się w czasie rzeczywistym zaistnienia zdarzeń), jak i *offline* (czyli sprawdzania archiwalnych zapisów w celu dotarcia do interesujących fragmentów zapisu z monitoringu).

1.2. Algorytmy przetwarzania sekwencji wideo

Poniżej przedstawiono ogólną koncepcję realizacyjną dla algorytmów przetwarzania sekwencji obrazów video. Zaprezentowano problematykę przechowywania oraz przetwarzania obrazów. Omówiono konwolucję dwu- i trójwymiarową jako typową technikę dla przetwarzania obrazów. Pokazano jej praktyczne zastosowanie przy wykrywaniu cech charakterystycznych dla obiektów. Zwrócono uwagę na szerokie możliwości zrównoleglenia algorytmów tego typu w praktyce oraz ich wykonywania w rozproszonym środowisku rozległej sieci komputerowej [1].

Nowoczesne algorytmy i techniki przetwarzania obrazów pozwalają na wszechstronną analizę oraz obróbkę obrazów nie tylko w zakresie typowych operacji, takich jak: skalowanie, zmiana kontrastu, filtracja w zakresie nasycenia, poprawa jakości i kolorystyki, etc., ale także w zakresie analizy i rozpoznawania obiektów oraz cech obiektów prezentowanych na obrazach. Wśród zastosowań można tutaj wymienić: lokalizację oraz śledzenie trajektorii obiektów ruchomych, nawigację dla ruchomych obiektów naziemnych i latających, robotykę przemysłową czy techniki radarowe oraz sonarowe. Istnieją generalnie dwa podejścia realizacyjne: w pierwszym projektuje się wyspecjalizowane układy scalone przeznaczone dla określonych algorytmów, w drugim stosuje się systemy oparte o zrównoleglenie obliczeń, a następnie scalanie ich wyników. Podejście drugie zapewnia większą elastyczność oraz znacznie większą przydatność dla prac naukowo-badawczych w zakresie testowania i weryfikacji nowych technik i algorytmów, a w szczególności pozwala na rozpraszanie obliczeń w sieciach rozległych. Pożądanym aplikacyjnie jest przetwarzanie w czasie rzeczywistym, gdzie przetwarzanie sekwencji obrazów wymaga, aby odpowiedź systemu komputerowego pojawiała się nie później niż nadejście kolejnej klatki z urządzenia rejestrującego [2].

1.2.1. Techniki przetwarzania obrazów – konwolucja

Pojedynczy obraz cyfrowy reprezentowany jest zazwyczaj w postaci maczyzy punktów w zadanej rozdzielczości. Każdy z punktów opisany jest poprzez informację o kolorze – w przypadku obrazów barwnych, lub stopniu szarości – w przypadku obrazów monochromatycznych. Dla reprezentacji obrazów barwnych stosuje się często reprezentację typu RGB, która wymaga przechowywania informacji oddzielnie o trzech barwach składowych. Inną reprezentacją jest tzw. *pseudo-kolor*, wówczas odpowiednie barwy są ponumerowane i identyfikacja z fizycznym kolorem odbywa się poprzez odwołania do tablicy wzorcowej (palety). Dla celów przetwarzania obrazów często stosuje się obrazy reprezentowane w skali szarości ze względu zarówno na jednowartościowy opis punktu na obrazie, jak i bezpośrednie powiązanie tej wartości z graficzną reprezentacją wyrażoną odcieniem szarości. W zakresie formatów dla zapisu ciągłego (np. fragment filmu) stosuje się (odpowiednio spakowaną) sekwencję klatek składowych.

Jedną z typowych technik przetwarzania (ogólnie pojętych) sygnałów jest konwolucja (ang. *convolution*) [3]. Konwolucja n -wymiarowa powstaje poprzez uogólnienie konwolucji $(n-1)$ -wymiarowej w n -ty wymiar. Idea konwolucji jednowymiarowej przedstawiona została w formule (1), gdzie I reprezentuje wektor numerycznych danych wejściowych, K to jednowymiarowy *kernel*, a O to wektor danych wyjściowych. Jak widać wektor danych wyjściowych O w (1) powstaje poprzez wymnożenie i zsumowanie kolejnych wartości sygnału z odpowiednimi wartościami kernela K .

$$O[n] = \sum_{i=1}^r K[i] \cdot I[n - i] \quad (1)$$

Poprzez matematyczną generalizację konwolucji jednowymiarowej w kolejny wymiar otrzymujemy konwolucję dwuwymiarową (2), która może już być wykorzystywana dla przetwarzania pojedynczych klatek czy obrazów.

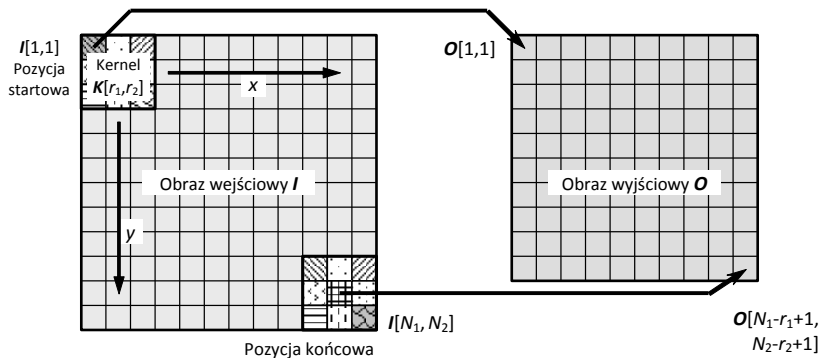
$$O[n_1, n_2] = \sum_{j=1}^{r_2} \sum_{i=1}^{r_1} K[i, j] \cdot I[n_1 - i, n_2 - j] \quad (2)$$

Po przekształceniach oraz przeskalowaniu indeksów otrzymujemy formuły (3) i (4), które mogą być zastosowane bezpośrednio w praktyce. Formuła (3) opisuje proces obliczania wartości dla piksela o współrzędnych $O[x, y]$. Formuła (4) opisuje cały obraz wyjściowy O . W praktyce, obraz wejściowy I jest reprezentowany poprzez macierz $N_1 \times N_2$ (np. o wymiarach rzędu 800x600) w odcieniach szarości o wartościach 0–255. Rozmiar kernela K w typowych zastosowaniach nie przekracza wielkości kilkunastu dla r_1 i r_2 . Jak pokazuje formuła (4) rozmiar obrazu wyjściowego O wynosi $N_1 - r_1 + 1$ na $N_2 - r_2 + 1$.

$$O[x, y] = \sum_{j=1}^{r_2} \sum_{i=1}^{r_1} K[i, j] \cdot I[x + i - 1, y + j - 1] \quad (3)$$

$$O = \left\{ O[x, y] \mid x \in \langle 1, N_1 - r_1 + 1 \rangle, y \in \langle 1, N_2 - r_2 + 1 \rangle \right\} \quad (4)$$

Konwolucja dwuwymiarowa posiada swoją interpretację graficzną, która przedstawiona została na rys. 1.1.

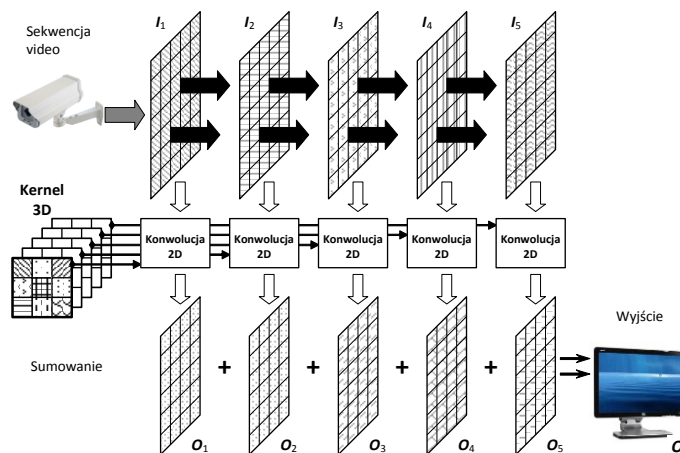


Rys. 1.1. Geometryczna interpretacja konwolucji dwuwymiarowej

Jak widać w formułach (3) i (4), na wynikową wartość dla piksela $O[x, y]$ mają wpływ wartości pikseli sąsiednich oraz wartości zawarte w kernelu K . Przy odpowiednim doborze rozmiarów r_1, r_2 oraz wartości dla kernela K wśród możliwych do realizacji technik przetwarzania wymienić można między innymi: poprawę kontrastu, wyostrenie lub rozmycie obrazu, detekcję krawędzi oraz wszelkiego rodzaju filtracje.

Z formuł (3) i (4) wynika, że konwolucja dla obrazu I to ciąg powtarzających się jednakowych i niezależnych od siebie operacji mnożenia i dodawania. Algorytmy wykorzystujące konwolucję nadają się do zrównoleglenia i rozproszenia obliczeń ze względu na względną niezależność poszczególnych operacji składowych. W rozwiązaniach równoległych zwykle wykonuje się rozproszenie pętli programowych zgodne z formułami (3) i (4), z rozbięciem obrazu I na podfragmenty i obliczaniem składowych konwolucji osobno dla każdego podfragmentu na niezależnej podjednostce

Konwolucja dwuwymiarowa dotyczy operacji na pojedynczych klatkach lub obrazach. Poprzez uogólnienie konwolucji dwuwymiarowej w dziedzinę czasową otrzymujemy trójwymiarową konwolucję przestrzenno–czasową. Geometryczna interpretacja tej konwolucji dla czasowego rozmiaru sekwencji wynoszącego 5 przedstawiona została na rys. 1.2. Przy wykorzystaniu konwolucji czasowo–przestrzennej można uzyskać różnego rodzaju filtracje polegające np. na eliminacji określonych obiektów z filmu video, generację kolejnej klatki na podstawie klatek poprzednich, etc. Dla tego typu algorytmów możliwości zrównoleglenia operują w trzech stopniach swobody. Podobnie jak dla konwolucji dwuwymiarowej, dwa pierwsze stopnie swobody dotyczą rozmiarów pojedynczej klatki $N_1 \times N_2$. Do tego dochodzi stopień swobody związany z długością sekwencji klatek [4].



Rys. 1.2. Interpretacja geometryczna konwolucji trójwymiarowej (przestrzenno-czasowej)

1.3. Analiza sekwencji video – przykład

Na rys. 1.3 pokazano przykład wykorzystania praktycznego w przetwarzaniu sekwencji obrazów (klatek). W przykładzie sygnał video podawany był z video, który odtwarzał uprzednio zarejestrowaną taśmę z nagraniem sytuacji rzeczywistej. Poszukiwany był wzorec białego jasnego samochodu – wzorec miał rozmiar 9x9 pikseli. Wzorec był korelowany (korelacja to konwolucja z odpowiednio zdefiniowanym kernelem) z obszarem poszukiwań, początkowo zlokalizowanym w miejscu, w którym rozpoczyna się droga (rys. 1.3 – lewy dolny róg). W chwili pojawienia się samochodu funkcja korelacji wzorca dawała wyraźne maksimum. W miarę poruszania się samochodu maksimum funkcji korelacji podążało za samochodem. Kolejne miejsca maksimum połączone zostały krzywą i w ten sposób powstał ślad trasy przejazdu. Należy również nadmienić, że wzorec samochodu był modyfikowany dynamicznie, tzn. uwzględniał zmiany obrazu samochodu w miarę jego przemieszczania się wzdłuż drogi. W przypadku zasłonięcia samochodu przez występujące słupy trakcyjne, maksimum funkcji korelacji wyraźnie słabło. Tym niemniej po ponownym pojawieniu się całej sylwetki samochodu można go było wyraźnie odnaleźć. Dla wartości maksimum funkcji korelacji eksperymentalnie dobrano wartość progową. Obszar poszukiwań początkowo umieszczony był w lewym dolnym rogu obrazu, a następnie po odnalezieniu wzorca, obszar ten podążał za samochodem, tak aby jego centrum pokrywało się z miejscem wystąpienia maksimum funkcji korelacji na poprzedniej klatce. Dla tego przykładu wyraźnie widać oczywiste możliwości zrównoleglenia obliczeń, np. poszukiwanie obiektu na poszczególnych klatkach osobno.



Rys. 1.3. Śledzenie obiektu obserwowanego nieruchomą kamerą

1.3.1. Wnioski

Powyżej przedstawiono konwolucję dwu- i trójwymiarową jako podstawową technikę przetwarzania obrazów. Pokazano wykorzystanie systemu przy śledzeniu obiektów ruchomych na obrazach rejestrowanych przez nieruchomą kamerę. Zasygnalizowano również rozmiar problematyki przetwarzania i analizy obrazów w przypadkach ogólnych. Pokazano szerokie możliwości zrównoleżenia przetwarzania oraz wykonywania rozproszonych obliczeń na niezależnych jednostkach, potencjalnie w sieci rozległej.

1.4. Wybrane zagadnienia implementacyjne

W praktycznych implementacjach systemów aktywnego monitoringu występuje wiele problemów. Poniżej opisano zagadnienia związane z różnorodnością formatów plików zawierających cyfrowe zapisy wideo oraz z algorytmami stosowanymi do ich przekształcania.

1.4.1. Formaty plików wideo

W praktycznych zastosowaniach wykorzystuje się szereg formatów dla plików zawierających cyfrowe zapisy multimedialne (wizyjne). Aktualnie stosowanych jest bardzo wiele formatów. Wynika to przede wszystkim z wprowadzania własnych standardów przez szeroką gamę producentów sprzętu wideo, jak również ze zorientowania tych standardów na określone zastosowania aplikacyjne.

Spśród rozmaitych formatów dla multimedii o charakterze przekazu wizyjnego można wymienić te najpopularniejsze:

- AVI – format plików wideo często wykorzystywany do zapisywania filmów obrabionych w programach do obróbki wideo. Jest często używany na urządzeniach mobilnych PDA. AVI jest odmianą formatu RIFF.
- MOV – technologia multimedialna rozwijana przez firmę Apple.
- MPG (MPEG) – format video obsługujący kodowanie MPEG-1 oraz MPEG-2.
- MP4 (MPEG-4) – kontener multimedialny. Oficjalne rozszerzenie pliku to .mp4, lecz w przypadku plików zawierających jedynie strumień dźwięku (np. AAC lub *Apple Lossless*) stosuje się również rozszerzenie .m4a.
- ASF – format plików wideo. Format ASF wykorzystywany jest najczęściej do przechowywania strumieni danych zakodowanych za pomocą Windows Media Audio (WMA) lub Windows Media Video (WMV).
- VCD – format zapisu cyfrowego strumienia audio–wideo na płycie kompaktowej.
- SVCD – standard nagrywania płyt bardzo podobny do VCD. Główną różnicą między tymi jest jakość obrazu.
- WMV (Windows Media Video) – format kompresji filmów.
- DV (Digital Video) – format cyfrowego zapisu wizji stosowany głównie w kamerach cyfrowych DVC (*Digital Video Camcorder*) oraz magnetowidach cyfrowych DVCR (*Digital Video Cassette Recorder*).
- FLV – format technologii Flash wykorzystywany w odtwarzaczach wideo na stronach internetowych np. YouTube. Plik .flv można odtworzyć za pomocą zwykłej przeglądarki internetowej z wtyczką Adobe Flash Player lub Gnash, a także osobnych programów, spośród których wiele udostępnianych jest bezpłatnie, np. Winamp, Moyea FLV Player, FLV Player, ALLPlayer (programy na licencji freeware), Mplayer, VLC media player (programy na licencji GPL).
- M2TS (*Blu-ray*) – format zapisu filmów HD przez kamery AVCHD.
- MKV – format przechowywania obrazu lub dźwięku w jednym pliku (kontener multimedialny).

W realizowanych aplikacjach dotyczących monitoringu obsługiwane są najpopularniejsze z zaprezentowanych standardów dla formatów video.

Zasadniczym komponentem jest tutaj biblioteka procedur przekształcających określone wybrane standardy do formatu umożliwiającego zautomatyzowane przetwarzanie numeryczne, czyli w praktyce konwersja do postaci opisującej poszczególne klatki przekazu na poziomie pojedynczych pikseli.

1.4.2. Przekształcenia kontekstowe

Operacje polegają na modyfikacji poszczególnych elementów obrazu w zależności od ich stanu i stanu ich otoczenia. Ze względu na rozmiar kontekstu mogą wymagać wielu powtarzalnych operacji, ale algorytmy są regularne i ponadto mogą być wykonywane na wszystkich punktach obrazu jednocześnie.

Filtry wykorzystywane do analizy obrazów zakładają, że wykonywane na obrazie operacje będą kontekstowe. Oznacza to, że dla wyznaczenia jednego punktu obrazu wynikowego trzeba dokonać określonych obliczeń na wielu

punktach obrazu źródłowego. Algorytm polega na obliczeniu funkcji, której argumentami są wartości pikseli z otoczenia. Otoczenie najczęściej utożsamiane jest z kwadratowym obszarem otaczającym symetrycznie aktualnie przetwarzany piksel [5].

Paradygmat przetwarzania:

master-slave,

single program multiple data.

Dane wejściowe

sekwencja klatek n klatek o rozdzielczości x na y, przy trzech składowych RGB.

Wynik

sekwencja n klatek o rozdzielczości x na y, przy zapisie w składowych RGB.

Typowe rozmiary danych

n – długość sekwencji, typowo 24 klatki na sekundę, typowa rozdzielczość: x=800, y=600, trzy składowe koloru RGB (3 bajty).

Nazwy instancji algorytmu

filtry liniowe, konwolucja (splot funkcji), filtry dolnoprzepustowe, filtry górnoprzepustowe (gradient Robertsa, maska Prewitta, maska Sobela), filtry górnoprzepustowe wykrywające narożniki, filtry górnoprzepustowe wykrywające krawędzie, etc.

1.4.3. Przekształcenia kontekstowe w dziedzinie czasu

Algorytmy operujące na trójwymiarowej reprezentacji obrazów, gdzie trzecim wymiarem jest czas, a w praktyce określona liczba występujących po sobie klatek w sekwencji. Algorytmy te dają możliwość przekształcania kontekstowego w dziedzinie czasu, gdzie klatka wynikowa jest rezultatem przetwarzania kilku (kilkunastu) sąsiadujących ze sobą klatek. Algorytmy te również nadają się do zrównoleglenia, przy czym ich złożoność jest tutaj trójwymiarowa.

Paradygmat przetwarzania

master-slave,

single program multiple data.

Dane wejściowe

sekwencja klatek n klatek o rozdzielczości x na y, przy trzech składowych RGB.

Wynik

sekwencja n klatek o rozdzielczości x na y, przy zapisie w składowych RGB.

Typowe rozmiary danych

n – długość sekwencji, typowo 24 klatki na sekundę, typowa rozdzielczość: $x=800$, $y=600$, trzy składowe koloru RGB (3 bajty).

Zastosowania

detekcja obiektów ruchomych, filtracja tła, estymacja ruchu obiektów.

1.5. Podsumowanie

Zaprezentowane algorytmy z dziedziny przetwarzania obrazów są możliwe do efektywnego wykorzystania w systemie Comcute. Podane algorytmy są w zdecydowanej wielkości w pełni skalowalne. Podstawowe cechy algorytmów, takie jak rozpraszanie obliczeń oraz rozpraszanie danych, są w pełni przystosowane do równoległej praktycznej realizacji (są skalowalne).

W dziedzinie przetwarzania obrazów danymi jest zazwyczaj sekwencja video składająca się z poszczególnych klatek. Każda z klatek jest reprezentowana poprzez macierz o określonych wymiarach (zależnych od jakości lub formatu), reprezentowana poprzez wartości pikseli (barwne lub nie). W przypadku typowego przetwarzania klatka po klatce od razu widać możliwości segmentacji danych na poszczególne klatki. Ponadto w ramach pojedynczej klatki wykonywane są operacje powtarzalne o rozmiarze kernela (rozmiar zdecydowanie mniejszy od rozmiarów klatki). Tutaj też istnieje możliwość segmentacji. W przypadku tym istotne jest rozważenie, czy rozproszenie wynikające z podzielenia będzie kompensowało zwiększone koszty komunikacyjne. Widać wyraźnie, że tego typu podejście znajduje uzasadnienie w praktycznych zastosowaniach dla systemu Comcute.

1.6. Wykaz literatury

1. Brudło P., Kuchciński K.: *Parallel Spatio-Temporal Convolution Scheme Oriented for Hardware Real-Time Implementation*, Proceedings: 23rd EUROMICRO Conference, Budapeszt, Węgry, wrzesień 1997, pp. 190-195
2. Brudło P.: *Wykorzystanie procesorów sygnałowych w przetwarzaniu obrazów*, Materiały konferencyjne: Przetwarzanie Sieciowe i Rozproszone, Jastrzębia Góra, listopad 1999, Praca zbiorowa Katedry Architektury Systemów Komputerowych KASKBOOK, pp. 155-161
3. Brudło P.: *System przetwarzania i analizy sekwencji obrazów video w oparciu o sieć procesorów sygnałowych serii ADSP-21060*, Materiały konferencyjne: Systemy Czasu Rzeczywistego '2000, Kraków, wrzesień 2000, pp. 485-494
4. Granlund G., Knutsson H.: *Signal Processing for Computer Vision*, Dordrecht, Holland, Kluwer Academic Publishers
5. Woźnicki J.: *Podstawowe techniki przetwarzania obrazu*, Wydawnictwa Komunikacji i Łączności, Warszawa